Ship Detection in SAR Images Based on Maxtree Representation and Graph Signal Processing

Philippe Salembier, *Fellow, IEEE*, Sergi Liesegang, and Carlos López-Martínez, *Senior Member, IEEE*

***Abstract*— This paper discusses an image processing architecture and tools to address the problem of ship detection in synthetic-aperture radar images. The detection strategy relies on a tree-based representation of images, here a Maxtree, and graph signal processing tools. Radiometric as well as geometric attributes are evaluated and associated with the Maxtree nodes. They form graph attribute signals which are processed with graph filters. The goal of this filtering step is to exploit the correlation existing between attribute values on neighboring tree nodes. Considering that trees are specific graphs where the connectivity toward ancestors and descendants may have a different meaning, we analyze several linear, nonlinear, and morphological filtering strategies. Beside *graph filters*, two new filtering notions emerge from this analysis: *tree* and *branch filters*. Finally, we discuss a ship detection architecture that involves graph signal filters and machine learning tools. This architecture demonstrates the interest of applying graph signal processing tools on the tree-based representation of images and of going beyond classical graph filters. The resulting approach significantly outperforms state-of-the-art algorithms. Finally, a MATLAB toolbox allowing users to experiment with the tools discussed in this paper on Maxtree or Mintree has been created and made public.**

***Index Terms*— Branch filter, graph filter, graph signal processing, machine learning, Maxtree, ship detection, support vector machine (SVM), synthetic-aperture radar (SAR), tree filter.**

I. Introduction

OCEAN and sea monitoring by means of Earth observation data include many activities and applications supporting different needs: sustainable fishing, marine ecosystems protection, natural resources extraction, commerce and trade, and so on. Undoubtedly, they are related as, for instance, marine fisheries around the world remain seriously threatened from fishing overcapacity. Marine ecosystems are polluted by activities related to the increase of maritime traffic. In the latest report of the state of the world's fisheries and aquaculture, the Food and Agriculture Organization of the United Nations reported that 87% of the world fishing stocks, for which assessment information is available, is either fully exploited (57%) or overexploited (30%), and only 13% is not fully exploited. In many occasions, overexploitation is conducted by illegal fishing involving small to medium size boats. Regarding maritime traffic, the world fleet of cargo-carrying vessels has increased from 77.500 vessels in 2008 to some 100.000 vessels in 2018 comprising a total capacity of more than 2.100 million deadweight tonnage. Consequently, maritime traffic governance, pollution control of maritime environments, monitoring of efficient, and sustainable fishing or of illegal fishing activities require reliable ship monitoring tools.

Synthetic Aperture Radar (SAR) has largely demonstrated in the past its capability to provide all-time all-weather data over large areas with a high spatial resolution. In the context of electronic systems, the high level of sophistication and autonomy of SAR systems is such that radar-based machine vision with SAR imaging heavily relies on machine intelligence. Nowadays, SAR capabilities are enhanced by a decrease in the revisit time from the SAR sensor constellations. As a result, ship detection tools based on SAR data have multiplied in the last decade. The importance of this type of applications is also supported by the commercial sector, as according to [1], the commercial remote sensing data market is dominated by optical data which represents 84% of that market. In recent years, SAR data has not experienced the same growth as optical data except for maritime surveillance including ship monitoring. Finally, maritime surveillance is expected to undergo a significant increase in the forthcoming years. For example, in the framework of the Copernicus program [2], maritime surveillance is expected to undergo an average annual growth rate of Copernicus benefits of 43% up to 2020.

Concerning the specific ship detection application based on SAR data, it has to be mentioned that as images refer to the radar return, they exhibit a very large dynamic range, often up to 60 dB and are moreover corrupted by the so-called speckle noise. As a result, many classical image processing techniques have limited performances in this context.

Generally, SAR ship detection algorithms are based on a constant false alarm rate (CFAR) approach. Nevertheless, the performances of a traditional CFAR technique decrease with high spatial resolution SAR data, as the clutter probability density function (pdf) presents a large variability and ships may have complex forms. This results in a high probability of a false alarm. In order to improve the detection of ships, many researchers have developed CFAR approaches that locally adapt to the intensity pdf considering Gaussian, Gamma, or more complex distributions. In [3], a lognormal mixture model is proposed to tackle the sea clutter pdf

Manuscript received January 31, 2018; revised June 5, 2018 and September 12, 2018; accepted October 9, 2018. This work was supported by the Spanish Ministerio de Economía y Competitividad and the European Regional Development Fund under Project TEC2013-43935-R and Project TEC2016-75976-R. *(Corresponding author: Philippe Salembier.)*

The authors are with the Signal Theory and Communications Department, Technical University of Catalonia–BarcelonaTech, Barcelona, Spain (e-mail: philippe.salembier@upc.edu; sergi.liesegang@upc.edu; carlos.lopez@tsc.upc.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TGRS.2018.2876603

variability. Nevertheless, more recently, techniques trying to take into account the data spatial correlation have been proposed. Leng *et al.* [4] propose a bilateral filter to include the spatial distribution of the SAR data. In [5], the data correlation is exploited, and in [6], the concept of superpixel is considered to assure a local homogeneous clutter. Among the CFAR techniques, the SUMO technique [7] is a pixel-based approach developed during the past 15 years under the aegis of the European Commission. Other approaches have also pursued a better characterization of the ship signal considering different ship features [8] or Haar-like features [9]. More recently, some authors have explored the idea to represent ships by ellipses [10]. Other approaches have also exploited the synthesis of independent lower resolution SAR images or look [11], [12], as only the ship signal maintains the correlation between the different looks. With respect to the evaluation of the detection performances, in some cases, the automatic identification system is considered, when available, as ground truth [13]. In this paper, we conclude that the detection performances depend on the SAR system, being the Radarsat-2 system in ScanSAR mode the one providing the best results. The technique proposed in this paper, based on the Maxtree representation, presents the advantage of avoiding the explicit need to define a given pdf for the sea clutter. Moreover, it also takes into account the data spatial relations and the ship geometrical characteristics. In this sense, this technique is able to combine different detection approaches considered separately in the past. Furthermore, as will be explained in the sequel, the approach avoids the use of any speckle filter and is, therefore, able to preserve the spatial resolution of the data. Finally, it also involves the use of a machine learning algorithm and therefore combines artificial intelligence ideas with SAR imaging systems.

As previously mentioned, we are interested in processing strategies that rely on tree-based representations of images. These representations provide a hierarchical description where nodes represent regions. As a result, they allow the detection to be based on both radiometric and geometric attributes. We believe that this feature together with the hierarchical aspect of the representation is important for many object detection tasks and in particular for ship detection. Most of the time, the first step of these strategies is to create a tree such as a Maxtree or Mintree [14], [15] (also known as Component Tree [16]), a Tree of Shapes [17] (also known as the Inclusion Tree [18]), a Binary Partition Tree [19] or an α-Tree [20]. In the past, they have been used in numerous applications such as object detection [21]–[23], attribute filtering [15], [24], [25], segmentation [19], [26], [27], texture analysis and image content retrieval [28], remote sensing [29]–[31], and visualization [32]. In the context of object detection, once the tree is constructed, most classical approaches compute radiometric and/or geometric attributes for each region represented by tree nodes and assign the resulting attribute values to these nodes. Then, the nodes and attribute values are individually analyzed and a statement about the presence of the object of interest is made. Instead of analyzing these attribute values individually, one contribution of this paper is to highlight that attribute values on neighboring nodes are correlated and that there is a potential interest in exploiting this correlation. The reason for this correlation is that the tree structure represents an inclusion relationship between regions: the region corresponding to a child node is included in the region corresponding to a parent node. As a result, the attribute values for child and parent nodes are related. To our knowledge, no studies have been reported on how to deal with this correlation.

A proper framework to deal with this attribute correlation in a tree is what has been recently called *graph signal processing*. Graph signals [33]–[35] are collections of data that live on a graph structure. In this paper, the data are radiometric or geometric attributes that have been evaluated on regions represented by the tree nodes and that are assigned to their respective nodes. They form a particular graph signal that we call *graph attribute signal*. A natural choice to deal with the attribute correlation could be to use graph signal processing tools such as linear graph filters as those defined in [36]–[38] or morphological graph filters [39]. However, these tools were defined for arbitrary graphs. In this paper, we are interested in trees which are specific graphs essentially representing hierarchies. In particular, the notion of connectivity toward ancestors and descendants may have a different meaning. We therefore propose processing tools that may take advantage of this distinction. As a result, the notions of *tree filter* and *branch filter* will be defined and discussed and their difference with *graph filter* highlighted. Linear, nonlinear (median filter), and morphological (erosion, dilation, opening, closing, and reconstruction) filters will be defined in both the tree and branch filtering contexts. Concerning tree attribute signal processing, an interesting proposal related to our work is made in [40]. It consists of processing a tree-based representation of images, a Maxtree, with a connected operator [25] that also relies on a Maxtree. However, the interest in [40] is essentially to extract the attribute extrema and not to study nor to propose processing strategies to deal with the attribute correlation in the tree structure.

Finally, we will present an architecture able to detect ships in SAR images relying on Maxtree representation and graph signal processing adapted to these representations. Several graph, tree, and branch filters will be evaluated in this context. Tree opening and branch mean filters will be shown to be particularly attractive. The experimental evaluation will also highlight the interest of going beyond graph filtering in the context of trees. Beside tree opening and branch mean filters, the detection scheme also involves a morphological Tophat relying on a tree reconstruction process and an extinction filter to process the node likelihood values. Finally, the resulting ship detection will be compared to four state-of-the-art techniques providing a significant improvement in detection performances.

This paper is organized as follows. Section II highlights the main processing architecture proposed for ship detection. Section III discusses how the Maxtree representation can be processed and how the results can be visualized. Many of these tools are used in Section IV for ship detection in SAR images. Finally, conclusions are reported in Section V.

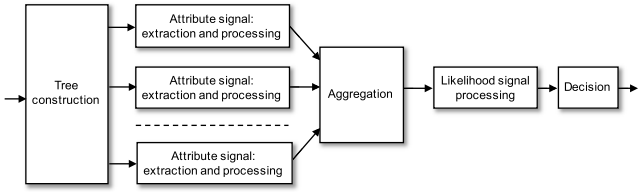


Fig. 1. Processing strategy for object detection.

II. Object Detection on Tree-Based Representation of Images

The tree-based object detection scheme we consider in this paper is illustrated in Fig. 1. It is a fairly generic scheme that can be used for many detection tasks but we will focus here on ship detection. The first step consists of computing a tree representation of the image, here a Maxtree [14]–[16]. A Maxtree describes the entire set of connected components resulting from the threshold decomposition of upper level sets. The resulting connected components are ordered by inclusion and structured in a tree. The tree leaves represent the image maxima and the root node the entire image support. A Maxtree can then be viewed as a multiscale description of the image maxima. This property makes them very attractive for ship detection given that ships appear as bright areas in SAR images. Note that the Maxtree is used here purely as an image representation which describes the original SAR image without any preprocessing. In this representation, the ship detection task is formulated as the search for tree nodes that correspond to ship instances.

To perform this search once the tree is computed, several attributes are extracted from the image. They can deal with radiometric as well as geometric information. This access to the geometric information is the second main motivation to use the Maxtree representation. The attributes are measured on the sets of pixels represented by the tree nodes and are assigned to their corresponding nodes creating *graph attribute signals*. Most object detection algorithms for tree-based representation of images analyze these attribute values individually for each node and make a prediction on the presence of the object of interest on the region of support corresponding to a tree node [22], [41]. However, as the tree represents inclusion relationships between regions, the attribute values between consecutive nodes along the tree branches are correlated. Taking advantage of this correlation is a key point of the strategy presented here and motivates the development of tools that consider the attributes populating the tree as graph attribute signals and process them to increase the detection robustness.

The scheme illustrated in Fig. 1 involves an aggregation step that takes into account the complete set of attributes and estimates the likelihood of each node to represent an instance of the object of interest, a ship. Several strategies can be used for this aggregation, and as suggested in [22], [41], one of the most efficient approaches relies on machine learning techniques. For example, if one has access to a training data set, a supervised classifier can be trained to estimate the object class probability. For the final decision, a straightforward approach can consider this likelihood as the final output and simply binarize it. However, as this likelihood populates the tree, it can itself be considered as a graph attribute signal which exhibits some correlation between consecutive nodes. As a result, the graph likelihood signal can be further processed to increase the robustness of the final decision.

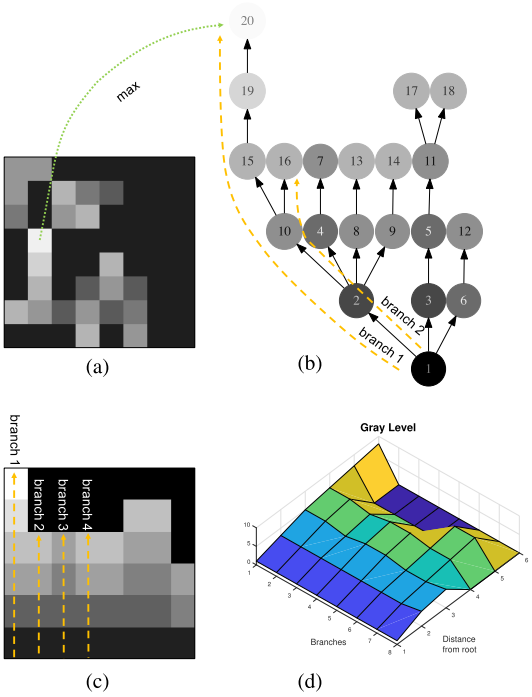


Fig. 2. Maxtree representation of images. (a) Original image. (b) Corresponding Maxtree. (c) 2-D branch representation. (d) 3-D branch representation.

Section III discusses the Maxtree representation, the creation of graph attribute signals for ship detection and the graph filtering tools needed to process these signals.

III. Maxtree Representation and Processing

A. Maxtree Representation and Visualization

A Maxtree represents all the binary connected components that can be extracted from an image by thresholding at all possible gray level values and it structures them by inclusion. More precisely, each tree node represents a connected component of the space that is extracted by the following thresholding process: for a given threshold T, consider the set of pixels X_T that have a gray-level value larger than or equal to T and the set of pixels Y_T that have a gray-level value equal to T. The tree nodes represent the connected components C of X_T such that $C \cap Y \neq \emptyset$. Note that a connected component obtained at a given threshold is included in another one obtained at a lower threshold. The links between the tree nodes represent this inclusion relationship between the connected components of X_T. Many algorithms have been proposed to compute Maxtrees. A review can be found in [42].

Fig. 2 illustrates the tree creation with a simple 8×8 image. The leaves of the Maxtree represent the image maxima. As the original image [Fig. 2(a)] has eight local maxima (assuming four connectivity), the corresponding Maxtree has

eight leaves [Fig. 2(b)]. The root node represents the entire image support (that is the connected component corresponding to the lowest possible threshold). The remaining nodes represent different connected components obtained at different thresholds. Finally, in Fig. 2(b), the gray levels used to display the nodes correspond to the threshold values that have generated the connected components.

The graphical representation of the tree, as shown in Fig. 2(b), is useful to visualize the structure of small trees. However, it is impractical to represent trees corresponding to images of reasonable size nor to analyze the evolution of graph attribute signals. An alternative representation that we have found useful in many situations is what we call *branch representation* shown in Fig. 2(c). It is a gray-level image in which the horizontal axis indexes the individual tree branches and the vertical axis corresponds to the distance of each node from the root. In the example of Fig. 2, the Maxtree has eight leaves, therefore eight branches, and the longest branch has length 6. Therefore, the branch representation is an 8×6 image. The gray-level value of pixel (i, j) in the branch representation corresponds to the attribute value (here the gray level) of the node that belongs to branch i and is at distance j from the root. Note that, as many nodes belong to multiple branches, they are duplicated in the branch representation. For example, the root node belongs to all branches. This is the reason why the gray level value associated with the bottom line of the branch representation is the same for all pixels. As the attribute values may drastically change between one branch to the next one, it is often useful to visualize the branch representation as a 3-D surface, as shown in Fig. 2(d).

Fig. 3(a) shows a typical image (3701×1651 pixels) which was obtained with the Radarsat-2 system over the Gulf of Guinea, western Africa. For visualization purpose, the logarithm of the actual gray-level values is shown. A small excerpt (200×200 pixels) of this image with three ships is presented in Fig. 3(b). This excerpt will be used for many illustrations in the following sections. The ships appear as areas of high mean gray-level values, with an elongated shape that could be approximated by an ellipse. The presence of speckle noise, inherent to SAR images, may hinder the detection. The Maxtree of this small excerpt involves 17.601 nodes distributed among 3.831 branches. As a result, a graphical representation as the one shown in Fig. 2(b) is not appropriate. Fig. 3(c) shows an alternative graphical representation of the tree with a scalable force directed placement (sfdp) algorithm [43] which efficiently deals with large graphs and which is available in the Graphviz software package [44]. The node gray-level values correspond to the gray levels of the image on the left. Note that many nodes and branches overlap in the drawing. The root node is in the center and an approximate idea of the tree structure can be inferred from this representation. However, the evolution of attribute signals is rather difficult to analyze. Therefore, the branch representation [Fig. 2(c) and (d)] will be used to visualize attribute signals in the sequel.

Once created, the Maxtree is populated with attributes that may indicate the presence of ships. We will use three attributes to characterize the binary connected components $\mathcal{C}$ associated with the Maxtree nodes: 1) the *mean gray-level* value of the pixels belonging to $\mathcal{C}$; 2) the *eccentricity* of the ellipse $\mathcal{E}$ that has the same second moments as $\mathcal{C}$[1]; and 3) the *area ratio* which is the relation between the area of $\mathcal{C}$ and that of $\mathcal{E}$. The intuition behind these attributes is that ships are represented by elongated ellipses of high mean gray-level values. Therefore, we are looking for bright areas with a high eccentricity and high area ratio (so that the shape of $\mathcal{C}$ is similar to an ellipse).

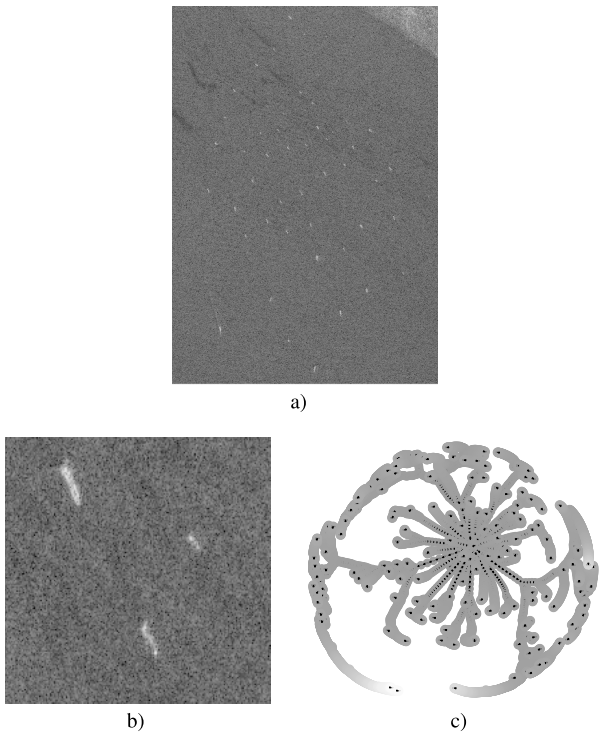


Fig. 3. Example of (a) original image (logarithm of gray levels) of the Gulf of Guinea, (b) 200×200 excerpt with three ships, and (c) Maxtree representation of the excerpt (Graphviz sfdp layout).

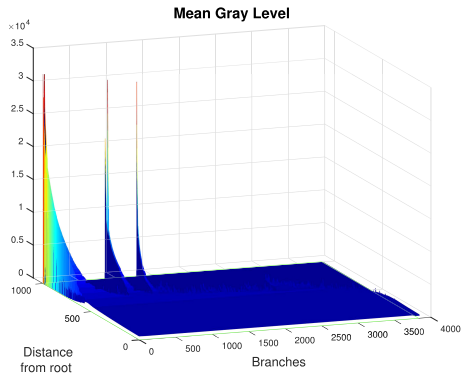


Fig. 4. Mean gray-level attribute (branch representation) for the Maxtree corresponding to the image of Fig. 3(b).

Fig. 4 presents the mean gray level attribute on the Maxtree of Fig. 3(b) (branch representation). The Maxtree involves 3831 different branches and the longest branch length is 1063. The branch representation is therefore an image of size 3831×1063. The presence of three branches on which the

[1]The eccentricity is defined here as the ratio of the distance between the two foci of the ellipse and its major axis length.

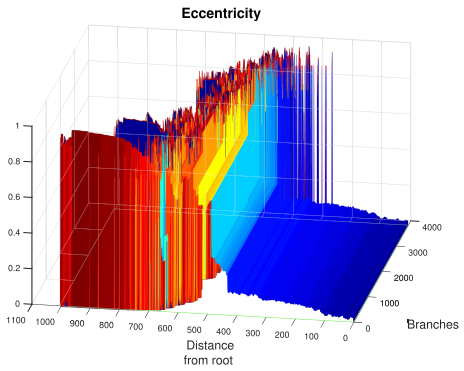


Fig. 5. Eccentricity attribute (branch representation) for the Maxtree corresponding to the image of Fig. 3(b).

mean gray levels reach extremely high values close to the leaves can be observed in Fig. 4. The mean gray-level attribute does not exhibit random variations and probably does not require any further processing. However, many attributes are not so clean and smooth. Let us consider the eccentricity attribute shown in Fig. 5. We can clearly observe the correlation in the attribute signal as values on neighboring nodes is most of the time related. However, we can also see random fluctuations along the tree branches in particular for sections close to the leaves. Random fluctuations that only affect locally very few nodes represent a challenge for the detection algorithm. They should be considered as noise and removed to increase the robustness of the detection. This kind of graph attribute signals motivates the creation of filtering tools as the one presented in the following section.

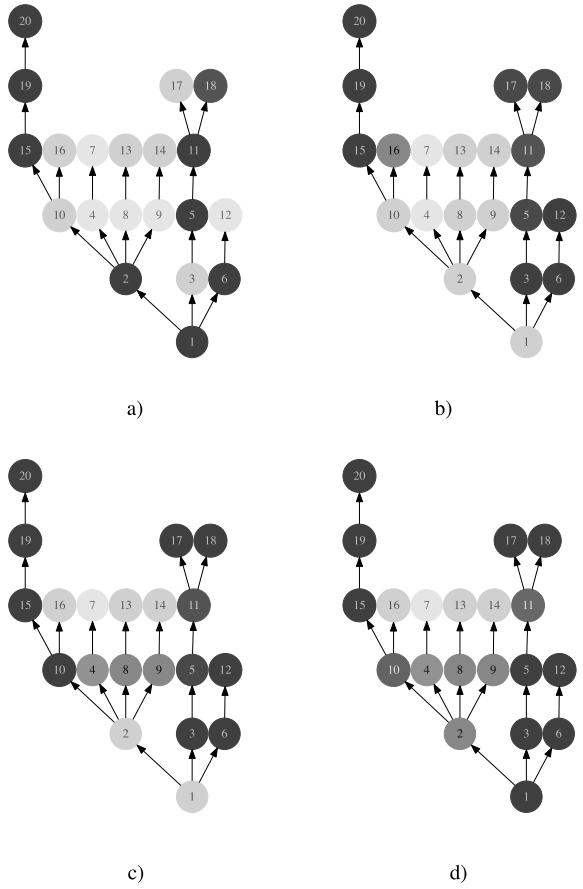


Fig. 6. Example of (a) attribute and the three types of filtering (b) graph, (c) tree, and (d) branch. In all cases, a median filter of size 2 is used.

B. Maxtree Filtering

1) Graph Filter: Graph signal filtering [39], [45]–[47] is generally considered as an extension of signal or image filtering. A popular approach to define them relies on the notion of *graph shift* [45] which can be considered as the equivalent of the classical time shift or delay. In the graph setting, it consists of replacing the signal values at a given node by a linear combination of values at its one-hop neighbors, which are specified by the adjacency matrix. Linear translation invariant filters are polynomials in the adjacency matrix. As a result, elementary linear graph filters replace the signal value at a node with a weighted linear combination of values of its K-hop neighboring nodes. A similar approach is used for morphological filter: a flat erosion (dilation) replaces the signal value at a node with the minimum (maximum) of values of its K-hop neighboring nodes. As usual in mathematical morphology, opening and closing are defined as combinations of erosion and dilation.

As an illustration, consider the example of Fig. 6(a). It corresponds to the Maxtree of the image of Fig. 2(a). It has been populated with attribute values chosen to illustrate various filtering approaches. Note that the attribute values are represented here by the gray levels used to display the nodes. Dark nodes represent low attribute values whereas bright nodes correspond to high attribute values. Assume that we want to apply a graph median filter [47] of size 2 to remove small fluctuations of the graph signal. The size of the median filter specifies which neighbors have to be included in the median computation. Here, size 2 means that all 2-hop neighbors, i.e., all nodes that are at a distance lower or equal to 2 from the current node have to be taken into account.[2] This is the classical *graph filter* illustrated in Fig. 6(b). Isolated attribute variations as the ones observed on nodes 3, 12, and 17 are then removed by the filtering.

For arbitrary graphs, beside the values of the adjacency matrix, no strong distinction among neighbors is generally made. However, trees are specific graphs representing hierarchies. Tree nodes have descendants and ancestors and the connectivity toward each one of them may play different roles, particularly in the context of object detection.

2) Tree Filter: Let us analyze node 10. In the context of the longest branch composed of nodes 1, 2, 10, 15, 19, and 20, the bright attribute value of node 10 can be considered as a small size fluctuation that should be removed by the median filter. However, with the graph median filter, the filtered value is determined by the values of node 10 itself, its descendant at distance 2: nodes 15, 16, and 19, its ancestors: nodes 2 and 1 and also the descendants of node 2, that are nodes 4, 8, and 9.

[2]In order to interpret the results of Fig. 6, note that when the median filter involves an even number of samples, the output is computed as the average of the two middle values.

In total, we have nine values, five bright values, and four dark ones. As a result, the filter output is bright. Note that many bright values come from nodes that do not belong to the same branch as node 10. If we want the filtering process to be more guided by the branch structure of the tree, we may prevent descendant nodes of the ancestors to influence the computation of the output value. This is the idea of the *tree filter* in which the neighborhood of a node is exclusively composed of all its descendants and all its ancestors that are at distance lower or equal to a given value. By comparison with graph filters, the K-hop neighborhood in tree filters is not defined by iteration of the adjacency matrix. The filtering process relies on two different shift operations: one shift toward ancestors and one toward descendants. Once the set of K-hop neighbors are defined by the two shift operations, linear filters, median filters, erosion, and dilation can be defined with rules similar to the ones used for graph filters. As usual, tree opening (tree closing) is defined as the composition of tree erosion (dilation) followed by tree dilation (erosion). They possess the algebraic properties of any opening (closing): increasing, idempotent, and antiextensive (extensive).

In the illustration of Fig. 6, consider the case of the tree median of size 2 applied on node 10, the neighborhood is defined by node 10, its descendants 15, 16, and 19 and ancestors 2, and 1. In particular, nodes 4, 8, and 9 which were involved in the 2-hop graph neighborhood are not considered anymore. The corresponding filtering results are shown in Fig. 6(c). As can be seen, the filtered value of node 10 is strongly influenced by this change of strategy.

3) Branch Filter: The third filtering approach deals with effects related to the disparity of node numbers in the different branches. A typical example can be seen in the root (1). Three main sets of branches are emanating from the root. Two of them (the one passing through nodes 3 and 6) involve mainly dark nodes. However, the filtered value of node 1 is largely dominated by node 2 and its descendants (nodes 10, 4, 8, and 9). Here, the key point is that the descendants of node 2 are much more numerous than the descendants of nodes 3 and 6. A possible way to deal with this issue is to use a *branch filter*.

It is a two steps filtering approach. In the first step, called *estimation* step, all branches passing through the current node to be filtered are extracted. Considering node 1 as an example, if a filter of size 2 has to be used, six individual branches are involved: they all start at node 1 and, respectively, ends at nodes 10, 4, 8, 9, 5, and 12. Note that if the node to be filtered is not the root node, the individual branches would also involve its ancestors. Once the branches are extracted, 1-D filters are individually applied on each of them. This estimation step produces several values that have to be *aggregated* in the second step. This filtering strategy has some similarity to the decomposition of 2-D separable filters into two 1-D filtering steps. This filtering framework is rich and many combinations of *estimation* and *aggregation* steps can be done. The corresponding study goes beyond the scope of this paper. Here, we have focused on four basic filters described in Table I. Moreover, we have defined the branch opening (closing) as the composition of a branch erosion (dilation) followed by a branch dilation (erosion). Note that an interesting filtering strategy may consist of using 1-D opening or 1-D closing as estimation step and then an aggregation step based on mean, median, min, or max (note that not all combinations produce opening or closing in the algebraic sense). We leave this study for future work.

TABLE I

DEFINITION OF THE ESTIMATION AND THE AGGREGATION STEPS FOR ELEMENTARY BRANCH FILTERS

Filter	Estimation	Aggregation
Branch Mean	1D Mean	Mean
Branch Median	1D Median	Median
Branch Erosion	1D Erosion	Min
Branch Dilation	1D Dilation	Max

As an illustration of the branch median filter, the filtering results can be observed in Fig. 6(d) and compared to the graph and tree filters results. Note that how the value at node 1 is less dominated by the high number of descendants of node 2.

In general, the three filtering approaches are different, but in some cases, the distinction disappears. Let us mention two examples as follows.

1) The graph and tree filters are equivalent for all filters of size one. The distinction appears for filters involving neighbors that are beyond 1-hop neighbors.
2) If branch erosion and dilation are defined as in Table I, in the case of flat structuring elements, they consist of computing, respectively, the minimum and the maximum of the input samples in a given neighborhood. As the neighborhood is the same for tree and branch filters, branch erosion (dilation) and tree erosion (dilation) are equivalent. This equivalence translates to their composition such as branch/tree opening and closing.

Fig. 7 illustrates the median filter of size 25 applied on the eccentricity attribute of Fig. 5. It can be observed how the filter has smoothed some of the attribute variations. The difference between the three filtering approaches is clearly noticeable for nodes that are at distance between 400 and 600 from the root node. In particular, in the cases of the graph and the tree filtering approaches, it can be observed that the eccentricity attribute values remain very high for these nodes although they were not so high in the unfiltered data. This phenomenon is caused by the disparity of node numbers in the different branches in particular for sections close to the leaves. By contrast, the branch filtering approach produces a smooth version of the signal with much less bias for the nodes at distance 400–600 from the root node. In Section IV-B, several graph, branch, and tree filters will be objectively compared in the context of the ship detection application.

C. Morphological Reconstruction on Maxtree

Morphological reconstruction is a classical image processing tool used to reconstruct extrema and create connected operators (See [25] and the references herein). It can also be applied on graph signals and defined through conditional dilations. The connectivity of the reconstruction is defined through an elementary structuring element $C_{\mathcal{G}}$ of size one.

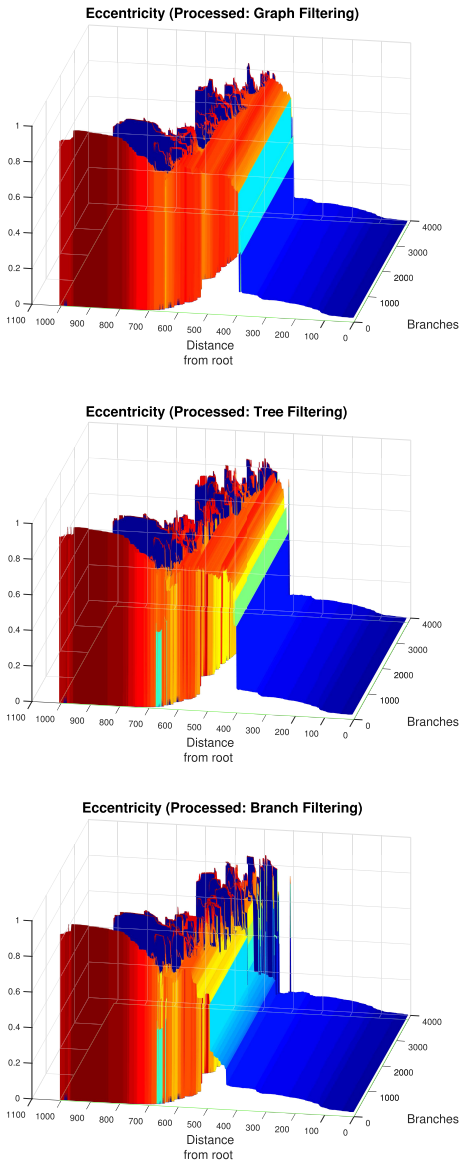


Fig. 7. Eccentricity attribute filtering with the three types of median filters (branch representation). (Top) Graph median filter. (Center) Tree median filter. (Bottom) Branch median filter.

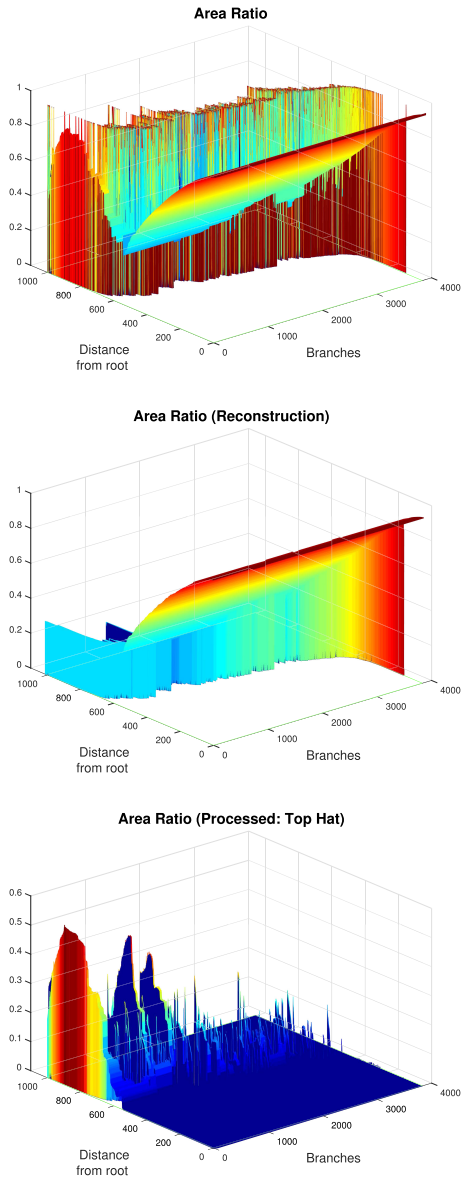


Fig. 8. Filtering of the area ratio attribute with a reconstruction from the root and a Tophat (branch representation). (Top) Original area ratio attribute. (Center) Reconstruction from the root. (Bottom) Tophat (after branch median filtering).

This structuring element includes all nodes that are at distance one from the current node. If f and g are graph signals (respectively, called the "reference" and "marker" signal), the antiextensive *graph reconstruction* $\rho^{\downarrow}_{\mathcal{G}}(g|f)$ *of* g *under* f is given by

$$g_k = \delta_{C_{\mathcal{G}}}(g_{k-1}) \bigwedge f$$
$$\text{and} \quad \rho^{\downarrow}(g|f) = \lim_{k\to\infty} g_k \tag{1}$$

where $g_0 = g \leq f$ and $\delta_{C_{\mathcal{G}}}$ is the graph dilation with $C_{\mathcal{G}}$.

By duality, an extensive *graph reconstruction* $\rho^{\uparrow}_{\mathcal{G}}(g|f)$ *of* g *above* f is given by

$$g_k = \varepsilon_{C_{\mathcal{G}}}(g_{k-1}) \bigvee f$$
$$\text{and} \quad \rho^{\uparrow}(g|f) = \lim_{k\to\infty} g_k \tag{2}$$

where $g_0 = g \geq f$ and $\varepsilon_{C_{\mathcal{G}}}$ is the graph erosion with $C_{\mathcal{G}}$.

Note that, as mentioned in Section III-B, erosion and dilation of size 1 are equivalent in the graph, tree, or branch approaches. Therefore, no distinction has to be made for the graph reconstruction. However, in practice, it may be useful to treat differently the connectivities toward descendants and ancestors. To this end, two new elementary structuring elements can be created: $C_{\mathcal{T}_C}$ and $C_{\mathcal{T}_P}$. These structuring elements include the current node and, respectively, all its children or its parent. Substituting $C_{\mathcal{G}}$ into 1 and 2 by either $C_{\mathcal{T}_C}$ or $C_{\mathcal{T}_P}$ creates two *tree reconstructions*. The reconstruction is from *root to leaves* with $C_{\mathcal{T}_C}$ and from *leaves to root* with $C_{\mathcal{T}_P}$. Depending on how the marker signal is defined, these two reconstructions can produce very different results. As classically done in mathematical morphology, reconstruction can be used to create a connected opening or closing [25]. Let us show an example.

Fig. 8 presents an illustration of reconstruction involving the area ratio attribute defined in Section III-A. It is the relationship between the area of the connected component represented by the node and that of its best fitting ellipse. The original area ratio attribute, Fig. 8 (top), is quite noisy. Therefore, a branch median filter of size 25 is applied. Furthermore, besides the noise, it can also be observed that the attribute values are rather high in the sections of the tree branches containing the ships (the same branches as the one highlighted in the mean gray-level attribute, see Fig. 4). Moreover, it is also very high for nodes close to the root. This is because the image support is a square and the best fitting ellipse is a circle. The regions corresponding to nodes close to the root are similar to the square image support and their area is close to the area of the best fitting circle. As the presence of this attribute maximum near the root does not reflect the presence of a ship, it is appropriate to remove it and this can be efficiently done with a reconstruction process. The original attribute signal plays the role of reference and the marker is a graph signal equal to zero everywhere except on the root node where it is equal to one. The results obtained after applying a tree reconstruction from root to leaves is shown in Fig. 8 (center). Note that, with this specific marker, a graph reconstruction would have produced the same result. The tree reconstruction is an opening and a tree Tophat is obtained by computing the difference between the original signal and the reconstructed one. This is shown in Fig. 8 (bottom). We see that the presence of ships is more clearly observable in the cleaned attribute signal.

D. Maxtree Extrema

Regional maxima (minima) are flat zones (largest connected component of constant value) whose gray-level value is higher (lower) than the surrounding flat zones. The distinction we have made in previous sections concerning the connectivity toward ancestors and descendants could be applied to the definition of extrema. It would lead to nodes that are considered as extrema or not depending on whether they are analyzed in the context of specific branches. It is unclear how this distinction could be useful in practice. As a result, we only consider the classical graph connectivity in which each node is connected to all its neighbors at distance one. In the context of trees, this includes all the children and the parent nodes.

Classical algorithms used to identify extrema in images can be easily adapted to graph and tree structures, but a more powerful way to deal with extrema is to follow the suggestion of [40]: use a Maxtree, a Mintree, or a Tree of Shape to describe the extrema of a tree-based representation of images. This idea is illustrated in Fig. 9. The original Maxtree, *Tree1*, is shown in Fig. 9(a). It has been populated with an attribute signal which exhibits three maxima. The attribute signal itself can be described by a second Maxtree, *Tree2*, presented in Fig. 9(b). The Maxtree construction is the same as the one classically used for images. The only difference is that the graph connectivity is used here. In Fig. 9, we have indicated the correspondence between the leaves of *Tree2* and the attribute signal maxima in *Tree1*. In Section III-C, we have mentioned that morphological reconstruction is one of the

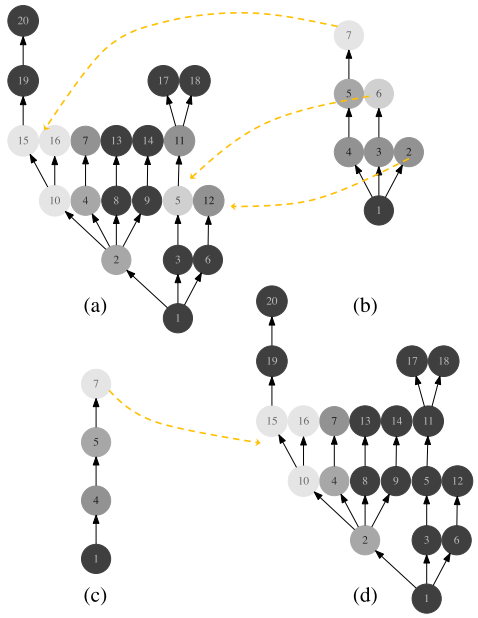


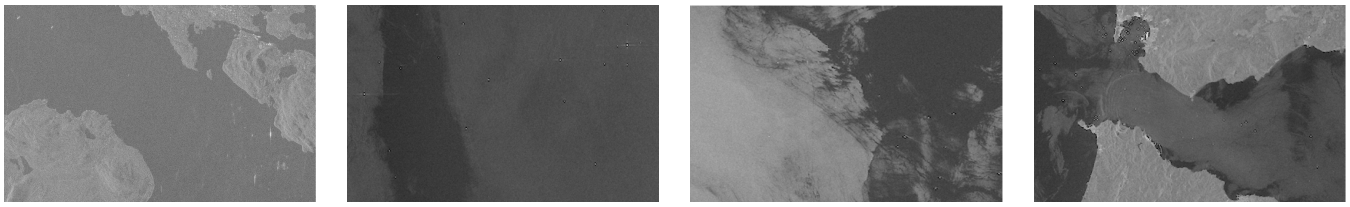
Fig. 9. Maxtree representation of the maxima of a Maxtree. (a) Original Maxtree, *Tree1*. (b) Maxtree representing the maxima of the original Maxtree, *Tree2*. (c) Pruned version of the Maxtree of Fig. 9(b) with an area criterion. (d) Restitution of the original Maxtree values with the pruned Maxtree of Fig. 9(c).

classical ways to build connected operators. An alternative strategy is to apply a pruning on a tree-based representation of the signal. We briefly recall this approach in the following section.

E. Maxtree Pruning

Pruning of Maxtree, Mintree, Tree of Shapes, or Binary Partition Tree has classically been used to simplify the representation itself or to create connected operators [14], [24], [26], [32]. This idea is illustrated in Fig. 9(c) for *Tree2*, which is the Maxtree describing the attribute signal of *Tree1*. The pruning consists in measuring a new attribute, the area in this example, and in removing all nodes of *Tree2* that have an area smaller than a given threshold, equal to two in this example. The nodes 2, 3, and 6 of *Tree2* have an area smaller or equal to two (they represent connected components of *Tree1* with two or less nodes), therefore, they have to be removed. The area is an increasing criterion so removing the nodes that do not meet the criterion directly defines a pruning. If the criterion is not increasing, several strategies have been proposed and more details about this issue can be found in [25] and its references.

Finally, the pruned Maxtree can be used to create a filtered version of the attribute signal of *Tree1*. Similar to what is done for images, the nodes of *Tree1* that were assigned to the pruned nodes of *Tree2* are now assigned to their first nonpruned ancestor. In the example of Fig. 9, this means that nodes 5, 11, and 12 of *Tree1* are now assigned to the root node of the pruned version of *Tree2*. In the tree restitution step shown in Fig. 9(d), the attribute value of nodes 5, 11, and 12 of *Tree1* are now restituted with the same attribute value as



a) b) c) d)

Fig. 10. Example of original images (logarithm of gray levels) of (a) Norway (Radarsat-2, Wide Fine Resolution mode) and (b)–(d) Gibraltar (Radarsat-2, ScanSAR Wide Beam mode).

the root node of *Tree2*. This is a connected operator, an area opening, applied on *Tree1*.

Before discussing how these ideas can be applied to ship detection, let us mention that we have created and published a MATLAB toolbox [48] involving most of the tools presented in this section allowing interested researchers to easily experiment with graph signal processing on Maxtree or Mintree.

IV. Ship Detection in SAR Images

The tools discussed in Section III can be used for many applications. In this paper, we are interested in their application to the problem of ship detection in SAR images. In the following, the proposed strategy used for this task is presented and its performance evaluated.

A. Attribute Definition and Processing on Maxtree

The detection strategy follows the scheme of Fig. 1. The Maxtree is initially created from the original image without any prior speckle filtering, preserving thus the original spatial resolution. To reduce the memory usage and the computational complexity, the tree may be simplified by removing all nodes that are either too small or too large to represent a ship. We refer to this step as area pruning. For example, taking into account the image resolution, all nodes involving less than 20 pixels may be removed from the Maxtrees. This leads to an average reduction of 60% in the number of tree nodes taking into account the images we process for this application. Moreover, considering the sizes of the world largest ships and the image resolution, nonroot nodes of size larger than 7.000 pixels may be considered as not relevant for ship detection and could also be removed. This leads to a further reduction of 10% of the tree sizes.

After the tree simplification, the three features used as an illustration in Section III are computed: 1) the *mean gray-level* value of the pixels belonging to each node $\mathcal{C}$; 2) the *eccentricity* of the ellipse $\mathcal{E}$ that has the same second moment as $\mathcal{C}$; and 3) the *area ratio* which is the relation between the area of $\mathcal{C}$ and that of $\mathcal{E}$. Note that, as the tree may have been simplified by removing the nodes that are too small or too large to correspond to ships, the detection algorithm actually relies on four attributes.

The mean gray-level attribute signal is left unprocessed. However, as discussed in Section III-C, the area ratio attribute signal exhibits high values for nodes corresponding to the ships, but also for the nodes close to the root. To remove the values related to the nodes that are close to the root, a tree reconstruction from root to leaves is applied and then used to compute a Tophat.

The following step is to use one of the filters discussed in Section III-B on the eccentricity and on the area ratio attribute signals. The goal of this filtering step is to smooth the attribute signals and remove noisy fluctuations to improve the detection robustness. In Section III, graph, tree, and branch median filters were used for illustration purposes but Section IV-B will objectively compare the graph, tree, and branch versions of the mean, the median, the opening, and the closing filters.

The next step is to aggregate the processed features and estimate the likelihood of each node to represent a ship. In [41], this aggregation is done following a marginal approach that considers all attributes as independent. We propose here to aggregate the attributes and to transform them into an estimate of the likelihood of ship presence through a multidimensional supervised machine-learning technique. To this end, a support vector machine (SVM) working as a binary classifier (ship class/nonship class) is used on individual nodes. We refer to this step as the *node classification* problem in the sequel.

To train and test the SVM, a node database was created. The database was extracted from two large images taken by Radarsat-2. Both images were acquired in the wide fine resolution mode. The first image can be seen in Fig. 3(a). It was acquired on March 18, 2013 and corresponds to the Gulf of Guinea, with a spatial resolution of 5.2 m in range and 7.7 m in Azimuth and an incidence angle about 42°. The size of image is 1651×3701 pixels. The second image is presented in Fig. 10(a) and was acquired on March 19, 2014. It corresponds to the coast of Norway, where the spatial resolution is 3.1 m in Azimuth and 4.6 m in range, presenting an average incidence angle on 25°. In this case, the image size is 3141×1979 pixels. As it can be observed, both images present similar spatial resolution. 50 images of size 100×100 pixels containing a variety of ships, sea, and land areas were extracted from the two large original images. To define the ground truth, we manually defined the *ideal ellipses* indicating the presence of ships and subjectively matching the ships contour. Then, we computed the Maxtree representation of the 50 images. In order to define the ground truth at the level of the tree nodes, we considered positive sample for the ship class, the Maxtree nodes with an overlap of at least 40% with the ideal ellipses. Negative samples were defined as nodes with no overlap with any ideal ellipses. Note that nodes with an overlap with ideal ellipses in-between

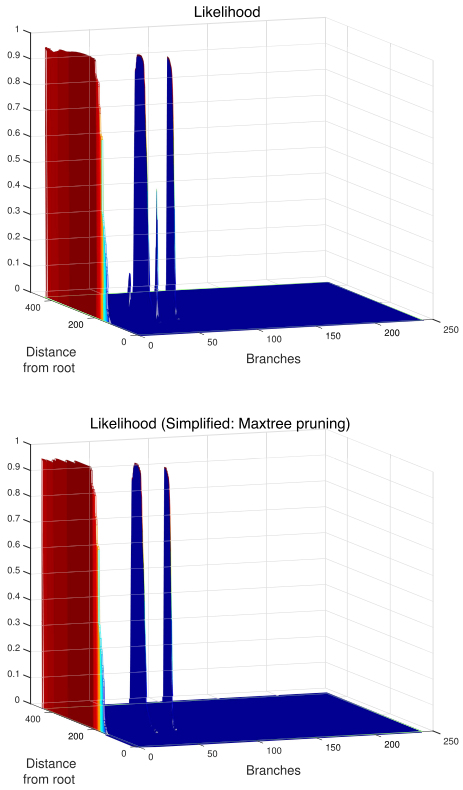


Fig. 11. Likelihood estimation (branch representation). (Top) Original likelihood. (Bottom) Processed likelihood.

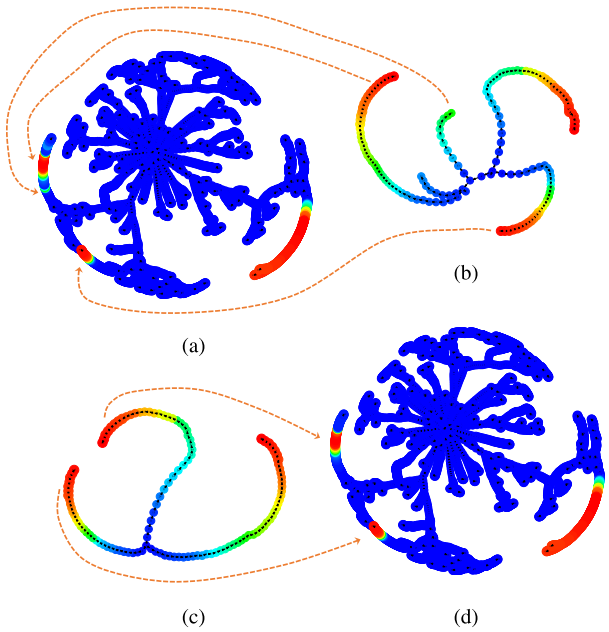


Fig. 12. Overview of the detection process (Maxtrees are represented with the sfdp layout of Graphviz). (a) Tree1: original Maxtree with likelihood attribute. (b) Tree2: Maxtree of Tree1 with the correspondence of some maxima. (c) Pruned Tree2 with a contrast extinction opening. (d) Restitution of the processed likelihood on Tree1.

0% and 40% were not considered in the node database. Following this procedure, we ended up with 4.000 ship nodes and 70.000 of nonship ones. Then, each class of the complete node data set was divided into training, validation, and test sets with a proportion of 60%, 20%, and 20%, respectively. The validation set was used to optimize the size of the filters.

Once trained, the SVM is used to populate the Maxtree nodes with a new attribute that is the ship class likelihood. The resulting likelihood attribute signal is shown in Fig. 11 (top). It corresponds to the same image as the one used in Section III to illustrate the processing tools. Note, however, that the support of the branch visualization is not exactly the same. In particular, the length and number of the branches have been drastically reduced because of the tree simplification based on the area pruning. The original number of branches (maximum branch length) was 3.831 (1.063) and is reduced to 241 (442) after size simplification.

The likelihood shown in the upper part of Fig. 11 clearly reveals the presence of three ships in three branches. This could be considered as the final detection result. However, it may be useful to consider this likelihood as a graph signal and further process it to remove spurious maxima that may hinder the detection in difficult cases. Although the results in the upper part of Fig. 11 are quite clean, we may observe more than three maxima. Some small maxima are visible on the middle branch beside the main large one.

Several processing strategies are possible but an efficient option relies on the Maxtree pruning discussed in Section III-E. As the goal here is to remove some of the maxima of the likelihood attribute signal, the first step is to compute its Maxtree representation. This step is illustrated in the upper part of Fig. 12. In this figure, *Tree1* represents the original Maxtree populated with the likelihood attribute signal. It corresponds to the sfdp layout of the branch representation shown in the top of Fig. 11. *Tree2* [Fig. 12(b)] is the Maxtree representation of the likelihood populating *Tree1*. We have indicated in the figure that the correspondence between three leaves of *Tree2* and three maxima of the likelihood attribute signal on *Tree1*. Note that the correspondence of the remaining two leaves and maxima is not drawn to maintain the figure clarity.

The complete removal of *Tree1* maxima corresponds to the pruning of entire terminal branches of *Tree2*. Here, terminal branches refer to complete sets of nodes going from a leaf node to a branch bifurcation. In the case of *Tree2*, we may want to remove all terminal branches of low contrast, i.e., sets of nodes such that the difference in likelihood values between a leaf node and a bifurcation node is lower than a given threshold. Following the suggestions made in [27], using different threshold values for this pruning would lead to hierarchical ship detection. Alternatively, we may want to remove terminal branches of small area that are terminal branches for which the area at a bifurcation point is lower than a threshold. This type of attribute is known as *extinction attribute* [49]. For an increasing attribute, the extinction value of a tree node is the maximal attribute value such that the terminal branch it belongs to still exists after the pruning. Extinction attributes can be efficiently computed on structures such as Maxtree [50] and essentially allow pruning of trees at nodes where a bifurcation exists. The tree resulting from an area extinction pruning is shown in Fig. 12(c). It perfectly preserves the three wide maxima and removes the presence of the remaining maxima.

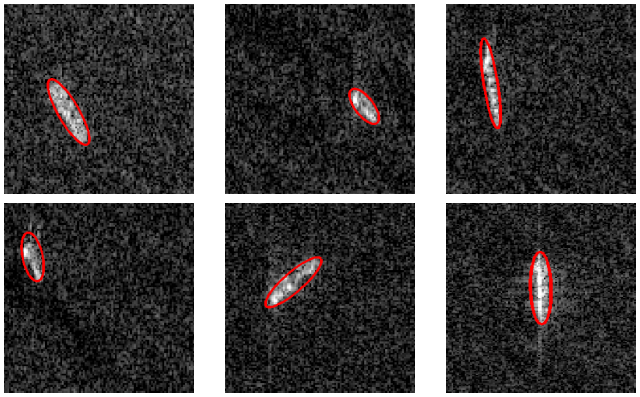


Fig. 13. Example of ship detection results (Radarsat-2, Wide Fine Resolution mode). The gray levels of the images have been adjusted to allow a clear view of the ship. Detections are indicated by red ellipses.

Finally, the simplified likelihood values can be restituted in the original *Tree1* [See Fig. 12(d) for the sfdp layout and the lower part Fig. 11 for the branch visualization]. The presence of three ships is perfectly highlighted by many nodes along three branches that have a high probability of being in the ship class.

The final output of the algorithm should be to state whether a ship is present or not at a particular image location. This last step is called here the *ship classification* task (which has not to be confused with the *node classification* task previously discussed). The idea is to keep all nodes with a ship class probability higher than a high value (e.g., 80%). Each of these nodes is modeled by its best fitting ellipse $\mathcal{E}$. Along each branch including the nodes with high probability, many nodes have a ship class probability higher than 80%. Each node is associated with an ellipse and the set of ellipses of nodes belonging to the same branch and having high ship class probability overlaps. However, in practice, only one ellipse per branch has to be kept. To this end, for each branch, we compute the median ellipse of the set of overlapping ellipses. This median ellipse is considered as the final output. Some detection examples are illustrated in Fig. 13. Note that with this strategy, we are able not only to detect the presence of a ship but also its orientation and, to some extent, its size.

In terms of evaluation, the *ship classification* task will be evaluated on the database composed of the 50 images already used for the *node classification* task. As previously mentioned, the ground truth in terms of the ship was specified by manually defining the ideal ellipses.

Before presenting the performances of the algorithm, let us come back to the statement made at the beginning of this section following which no speckle filter is used. We believe that this is an important point as it allows us to preserve the image resolution which is essential for the detection of small objects as ships in SAR images. The speckle noise is often modeled as a multiplicative noise. It generates positive and negative fluctuations around the pixel noise-free value. In the Maxtree, the positive fluctuations are represented as sets of nodes forming small branches. The negative fluctuations are not represented on their own by specific nodes. They appear in nodes combined with other pixels of the same gray-level value. It is the SVM node classification step that handles the presence of the speckle noise. To be more precise, the SVM learns to distinguish between nodes representing ship instances and all other types of nodes including nodes representing the sea clutter, nodes representing positive fluctuations of the speckle noise, nodes combining pixels corresponding to negative fluctuations of the speckle noise with other pixels, nodes corresponding to ground areas, and so on.

TABLE II
NODE CLASSIFICATION RESULTS WITHOUT FILTERING

Set	Precision	Recall	F-Score
Training	0.853	0.875	0.864
Validation	0.840	0.868	0.853
Test	0.828	0.892	**0.858**

B. Evaluation

This section discusses the objective evaluation of the tools presented in previous sections and also assesses the ship detection performances. The proposed algorithm involves two classification steps: one dealing with nodes and the final one dealing with the ship. Let us start by evaluating the node classification step.

1) Node Classification: In the context of the node classification, an SVM with a Gaussian kernel is trained. As classically done for binary classification, the performance of the algorithm is assessed with the Precision, P, and Recall, R, parameters as well as $F_{\text{Score}} = 2\ PR/(P+R)$.

Let us start by providing the reference classification results obtained with a system where no area pruning is done and no filter is used on the attribute signals. The only processing which is done is the tree Tophat applied on the area ratio (to remove the high values of the attribute on nodes close to the root). Table II gives the Precision, Recall, and F-Score of the node classification for the training, validation, and test sets. As the values are very similar for the three sets, we may conclude that there is no overtraining or overfitting of the SVM.

The next step is to evaluate the interest of applying a filter on the area ratio and the eccentricity attribute signals. Table III provides the results for the graph, tree, and branch versions of the mean, the median, the opening, and the closing filters. Note that we do no evaluate erosions nor dilations as they would not preserve the position of the transitions in the attribute signals. Moreover, the results for the tree opening and closing are the same as the ones for the branch opening and closing. As discussed in Section III-B, these can be viewed as alternative implementations of the same filter. The F-Score evaluated on the test set can be considered as the final estimation of the system performances. Therefore, in Table III, we have highlighted in bold font the best F-Score results on the test set (values above 0.930). They correspond to the tree/branch opening and the branch mean filters. Note in particular how these values are significantly higher than the one obtained with the reference system that does not involve any filter (Table II). Based on these results, we have selected the tree/branch opening for the remaining experiments

TABLE III
INFLUENCE OF ATTRIBUTE SIGNAL FILTERING ON THE NODE CLASSIFICATION RESULTS WITH SVM

	Graph filters			Tree filters			Branch filters		
Mean	**Precision**	**Recall**	**F-Score**	**Precision**	**Recall**	**F-Score**	**Precision**	**Recall**	**F-Score**
Training	0.918	0.861	0.889	0.887	0.845	0.865	0.927	0.922	0.925
Validation	0.915	0.866	0.890	0.867	0.840	0.853	0.912	0.924	0.918
Test	0.913	0.874	0.893	0.866	0.843	0.854	0.926	0.934	**0.930**
Median	**Precision**	**Recall**	**F-Score**	**Precision**	**Recall**	**F-Score**	**Precision**	**Recall**	**F-Score**
Training	0.884	0.872	0.878	0.879	0.906	0.892	0.897	0.919	0.908
Validation	0.863	0.869	0.866	0.875	0.905	0.889	0.897	0.931	0.914
Test	0.887	0.890	0.888	0.876	0.910	0.893	0.891	0.930	0.910
Opening	**Precision**	**Recall**	**F-Score**	**Precision**	**Recall**	**F-Score**	**Precision**	**Recall**	**F-Score**
Training	0.905	0.881	0.893	0.924	0.943	0.933	0.924	0.943	0.933
Validation	0.906	0.874	0.890	0.922	0.946	0.934	0.922	0.956	0.934
Test	0.886	0.871	0.878	0.920	0.953	**0.936**	0.920	0.953	**0.936**
Closing	**Precision**	**Recall**	**F-Score**	**Precision**	**Recall**	**F-Score**	**Precision**	**Recall**	**F-Score**
Training	0.844	0.861	0.872	0.897	0.908	0.902	0.897	0.908	0.902
Validation	0.887	0.871	0.879	0.890	0.927	0.908	0.890	0.927	0.908
Test	0.867	0.866	0.866	0.882	0.928	0.905	0.882	0.928	0.905

TABLE IV
NODE CLASSIFICATION RESULTS: INFLUENCE OF THE TREE TOPHAT AND THE AREA PRUNING (RADARSAT-2, WIDE FINE RESOLUTION MODE)

	Tree/Branch Opening without Tree Tophat without area pruning			Tree/Branch Opening with Tree Tophat without area pruning			Tree/Branch Opening with Tree Tophat with area pruning		
	Precision	**Recall**	**F-Score**	**Precision**	**Recall**	**F-Score**	**Precision**	**Recall**	**F-Score**
Training	0.908	0.926	0.917	0.924	0.943	0.933	0.991	0.986	0.988
Validation	0.904	0.929	0.916	0.922	0.946	0.934	0.993	0.990	0.991
Test	0.896	0.931	0.913	0.920	0.953	0.936	0.995	0.993	0.994

presented in this paper. This experiment allows us to draw two conclusions: first, the use of filters applied on graph signals actually improves the node classification performances. Second, in the context of tree, there is an interest in going beyond graph filters as the best results are provided by tree or branch filters.

The last question we address concerns the influence of the Tophat and of the area pruning. The central part of Table IV reports the results of the tree/branch opening with the tree Tophat on the area ratio and without the area pruning. Those are the same results as the one given in Table III. On the left side of Table IV, one can see the decrease in performances in case the tree Tophat is not used on the area ratio. Finally, on the right side of the table, the results obtained when the area-based tree pruning is done. The increase of performances in both precision and recall reveals that the presence of false positives and false negatives is still significant for nodes corresponding to small areas. Therefore, the area pruning is useful not only to decrease the tree complexity but also to improve the classification results. As a conclusion, the final node classification algorithm involves the area pruning, the tree Tophat, and the tree/branch opening. The resulting node attribute values are used as descriptors by the SVM which in turn estimates the node likelihood to represent a ship.

2) Ship Classification: Once the node likelihood has been estimated with the SVM, the last step of the algorithm is to detect the presence of ships. To increase the robustness of this last step, we have proposed to use an extinction filter applied on the likelihood attribute signal.

TABLE V
FINAL SHIP DETECTION PERFORMANCES WITH AND WITHOUT THE AREA EXTINCTION FILTER ON THE LIKELIHOOD ATTRIBUTE SIGNAL. COMPARISON WITH STATE-OF-THE-ART APPROACHES [3], [11], [12], [51] (RADARSAT-2, WIDE FINE RESOLUTION MODE)

Ship detection approach	Precision	Recall	F-Score
Proposed approach without area extinction filter	0.878	1.000	0.935
Proposed approach with area extinction filter	0.947	1.000	**0.973**
CFAR approach [3]	0.865	0.889	0.877
Wavelet-based approach [11]	0.857	0.923	0.889
GLRT approach [12]	0.872	0.944	0.907
Entropy-based dissimilarity [51]	0.800	1.000	0.889

To evaluate the ship detection performances, we considered as true positive the cases where the final ellipse produced by the algorithm estimating the presence of a ship had an overlap of at least 40% with the ideal ellipse. Table V shows the ship detection performances with and without the area extinction filter. As can be seen, the area extinction filter provides a clear improvement in terms of Precision. The final F-Score on the ship detection task is very high.

Finally, we compare these results with four state-of-the-art techniques for ship detection in SAR images [3], [11], [12], [51]. The algorithm proposed in [3] is a CFAR algorithm that essentially relies on the gray-level attribute. The approach described in [11] relies on the discrete wavelet transform and deals with spatial pixel correlation at multiple resolutions.

In the case of [12], the strategy consists in detecting coherent targets based on the generalized likelihood ratio test (GLRT) initially developed in [52]. Finally, Wang and Chen [51] propose to measure the local dissimilarity between ships and their neighborhood by using a variance weighted information entropy measure. The results are reported in Table V. Note that, as our algorithm involves a pruning step removing all regions of areas smaller than 20 pixels, the output of the state-of-the-art algorithms [3], [12], [51] was also postprocessed by removing all detections involving less than 20 pixels (it was not possible to do this postprocessing with [11] as the algorithm provides the locations of ship detections but not a binary mask of the detected areas). With [12], [51], we have also noticed that a single ship could sometimes be represented by several connected components. As a result, we have added a morphological opening with a small structuring element in order to reconnect these components. The size of the opening was chosen such that it maximizes the F-score. The state-of-the-art algorithms perform rather well. Among them, the GLRT algorithm provides the best results. Finally, our proposed algorithm provides a significantly higher F-score than the four state-of-the-art algorithms.

Once trained, the scheme proposed in this paper can be used on any image that is similar to the ones used during the training stage. However, if there is a strong mismatch between the images to process and the ones used for the training, a new training may be necessary. In any case, the results quality reported here on highresolution images acquired by Radarsat-2 in Wide Fine Resolution mode are not dependent on this particular mode or sensor. To illustrate this point, we have created a second database composed of images also acquired by Radarsat-2 but on August 4, 2013 with the ScanSAR Wide Beam mode. They can be seen in Fig. 10(b)–(d). They correspond to the strait of Gibraltar. The resolution of these images is of 50 m both in Azimuth and range. It is much lower than the ones obtained with the Wide Fine Resolution mode. The incidence angle is between 20° and 49° and the size of the resulting images is 1700×1100 pixels. We followed exactly the same procedure as the one described for the Wide Fine Resolution case [50 images of size 100×100 pixels containing a variety of ships, sea, and land areas were extracted. The ground truth was manually defined. The node data set was divided into training (60%), validation (20%), and test (20%) sets]. As the resolution of these images is very different from the one corresponding to the Wide Fine Resolution mode, the features characterizing the geometry of the data have significantly different statistics. As a result, a new training of the SVM for the node classification was performed. In fact, any change of sensors or of acquisition mode that implies significant changes in the radiometric or in the geometrical characterization of the image content requires a new SVM training. Note that, as the resolution of the image is rather low compared to the ship size, we have not used any size-oriented tree pruning removing nodes corresponding to small areas. This is the only modification that was made to the algorithm. The results of the node classification step are shown in Table VI. We only reproduce here the results for the branch opening which was found to be the best in the Wide Fine Resolution mode. The final ship detection results are given in Table VII. As can be seen, the F-Score for node as well as ship detection are very good and the results are similar to the ones previously reported for Wide Fine Resolution images. Finally, some detection examples are illustrated in Fig. 14.

TABLE VI

NODE CLASSIFICATION RESULTS FOR THE STRAIGHT OF GIBRALTAR IMAGES (RADARSAT-2, SCANSAR WIDE BEAM MODE)

	Tree/Branch Opening		
	Precision	**Recall**	**F-Score**
Training	0.994	0.986	0.990
Validation	0.994	0.987	0.991
Test	0.996	0.989	0.993

TABLE VII

FINAL SHIP DETECTION PERFORMANCES FOR THE STRAIGHT OF GIBRALTAR IMAGES (RADARSAT-2, SCANSAR WIDE BEAM MODE)

Precision	**Recall**	**F-Score**
0.968	1.000	0.984

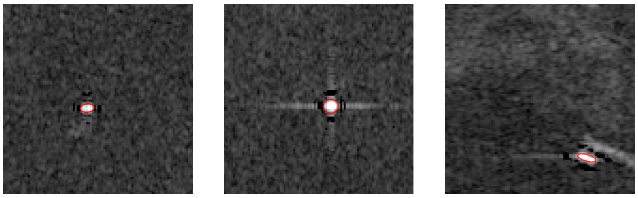


Fig. 14. Example of ship detection results (Radarsat-2, ScanSAR Wide Beam mode). The gray levels of the images have been adjusted to allow a clear view of the ship. Detections are indicated by red ellipses.

Compared to classical SAR-based ship detection algorithm, the approach described in this paper presents some important conceptual advantages. First of all, classical techniques are usually based on the use of a speckle filter in order to minimize the influence of this noise component. This filtering step induces a loss of spatial resolution, which is critical in the ship detection application, as the objects of interest correspond to very small areas involving a very low number of pixels. Instead, the proposed approach does not need any preliminary filtering step and, thus, avoids the loss of spatial resolution and does not alter the ship shape. Moreover, the proposed technique has an additional advantage which is that the detection can be based not only on radiometric features, as most conventional approaches, but also on geometric features. Here, eccentricity and the area ratio are two attributes that characterize the geometry of the regions associated with the tree nodes. In addition, the use of graph filtering tools adapted to the tree structure allows us to increase the detection robustness. This improvement results from the exploitation of the correlation existing among the attributes in neighboring nodes. To support this last statement, we have run a final experiment on the Radarsat-2 Wide Fine Resolution mode images that consisted in removing the tree/branch opening as well as the tree Tophat involved in the node classification stage. The ship classification step remained unchanged (in particular involving the area extinction filters). This experiment revealed

that the F-score dropped from 0.973 (as shown in Table V) to 0.864. This highlights the importance of the graph filtering associated with the node classification in the context of the complete algorithm. Finally, another advantage of the proposed technique is that it does not require to explicitly define a sea clutter pdf nor to process differently sea and ground areas.

In terms of limitations, we may note that with the current MATLAB implementation we have used for the experiments, the algorithm is not very CPU efficient. The Appendix provides some information about the CPU workload we have observed during the experiments and its distribution. It also includes some hints on how to improve the CPU efficiency. A second area of improvement could address the processing of unfocused ship response caused by ship motion. Finally, as previously stated, our approach has not processed differently sea and ground areas. We have not observed any significant reduction of robustness in ground areas but if, in the context of a specific application, one wishes to remove ground areas from the processing, ground masks, or coastal line detection algorithms such as [53]–[55] could be used.

Finally, the results reported in this paper were obtained with the MATLAB Maxtree Processing toolbox [48]. This package was designed to allow easy experimentation of most of tools described in this paper.

V. Conclusion

This paper has discussed an image processing architecture and a set of tools to address the problem of ship detection in SAR images. The detection strategy relies on a Maxtree representation of images and graph signal processing tools. Radiometric as well as geometric attributes are evaluated and associated with the Maxtree nodes forming graph attribute signals which are further processed to take into account the correlation between neighboring attribute values in the tree.

Considering that trees are specific graphs where the connectivity toward ancestors and descendants may have a different meaning, we have analyzed several linear, nonlinear, and morphological filtering strategies. Beside *graph filters*, two new filtering notions have emerged: *tree* and *branch filters*. The graph filter approach relies on the classical graph connectivity where no distinction is made between ancestors and descendants. The tree filter approach limits the neighborhood to the descendants and the ancestors of the node to be filtered and removes the influence of all remaining descendants of the ancestors. As a result, the transfer of information from one branch to another is limited. Finally, the branch filter controls the effects related to the disparity of node numbers in the different branches. It is a two steps filtering approach: In the estimation step, all branches passing through the current node are filtered and produce several values that are aggregated in the second step. Beside these filters, morphological reconstruction and connected filters on tree-based representation of an image are also discussed.

The complete ship detection algorithm involves these graph signal filters applied on the Maxtree representation. The processed attribute values are used with machine-learning tools (SVM) to detect ships. The proposed algorithm demonstrates the interest of applying graph signal processing tools on tree-based representation of images and of going beyond classical graph filters. The resulting ship detection approach significantly outperforms state-of-the-art algorithms.

In terms of the future work related to the ship detection application, one may consider the use of deep-learning techniques avoiding the use of hand-crafted attributes. However, this approach would require a large database of SAR images suitable for ship detection and its associated ground truth. As such a database does not currently exist, its creation would be the first challenge to be faced.

Note that the graph processing tools have been applied in this paper on Maxtree, but they can be applied on any tree-based representation of images including Mintree, Tree of Shapes [17], Binary Partition Tree [19], or α-Tree [20]. The use of these trees and associated graph processing tools for other applications than ship detection is an attractive topic. In term of extension of our work, depending on the application, one could extend the set of attributes used to train the classifier and include for example texture attributes such as those described in [56], [57] or even learned attributes as proposed in [58].

In the context of branch filters, interesting research could certainly be done on a complete study of combinations of estimation and aggregations steps.

Finally, a MATLAB toolbox involving most of the tools presented in this paper allowing interested researchers to easily experiment with graph signal processing on Maxtree or Mintree has been made public [48].

TABLE VIII
CPU Time Distribution

Processing step	Time %
Maxtree creation	0.24
Attribute computation	
Area	0.55
Mean gray level	1.27
Ellipse estimation	64.47
Eccentricity	0.19
Area ratio	0.46
Attribute signal processing	
Area pruning	2.16
Tree/branch opening on eccentricity	14.44
Tree/branch opening on Area ratio	14.49
Tree Tophat	0.39
SVM classification	0.09
Likelihood processing	
Maxtree of Maxtree creation	0.97
Extinction filter	0.003
Median ellipse computation	0.02

Appendix

With the MATLAB implementation we have used for the experiments of this paper, the processing time of ship detection on a 100×100 image is equal to 15 s in average. The distribution of this CPU time is described in Table VIII. As can be seen, more than 64% of the time is devoted to the computation of the best fitting ellipse of the Maxtree nodes. This step relies on the MATLAB `regionprops` command. Besides this step, the tree/branch opening applied on both the

TABLE IX
RELATIVE CPU TIME FOR THE GRAPH, TREE, AND BRANCH FILTERS. THE GRAPH MEAN FILTER IS USED AS REFERENCES. ALL FILTERS HAVE LENGTH 25

	Graph filter	Tree filter	Branch filter
Mean	1.00	0.36	0.87
Median	0.98	0.40	0.83
Opening	2.34	0.84	1.36
Closing	2.70	0.81	1.46

eccentricity and the area ratio attribute signal represents each about 14% of workload. A proper C++ implementation and a research on the best algorithms to perform these steps should provide a more efficient algorithm.

Table IX describes the relative workload of the various filters involved in the experiments. We have used as reference the graph mean filter of size 25. This table shows that, with the current implementation, tree filters are much more efficient in terms of workload than graph filters and that branch filters are slightly more efficient than graph filters. Moreover, mean, median, erosion, and dilation need roughly the same CPU time whereas opening and closing need twice as much time as they involve a combination erosion and dilation.

ACKNOWLEDGMENT

The authors would like to thank Prof. J. Mallorqui and Dr. M. Sanjuán-Ferrer for providing results of the algorithm defined in [11], [12] and Prof. Y. Cui for providing the software corresponding to [3].

REFERENCES

[1] G. Denis *et al.*, "Towards disruptions in Earth observation? New Earth Observation systems and markets evolution: Possible scenarios and impacts," *Acta Astronautica*, vol. 137, pp. 415–433, Aug. 2017.

[2] *Copernicus Market Report*, vol. 1, Pub. Office Eur. Union, PriceWaterhouseCoopers, London, U.K., Nov. 2016.

[3] Y. Cui, J. Yang, and Y. Yamaguchi, "CFAR ship detection in SAR images based on lognormal mixture models," in *Proc. 3rd IEEE Int. Asia–Pacific Conf. Synth. Aperture Radar*, Seoul, South Korea, Sep. 2011, pp. 1–3.

[4] X. Leng, K. Ji, K. Yang, and H. Zou, "A bilateral CFAR algorithm for ship detection in SAR images," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 7, pp. 1536–1540, Jul. 2015.

[5] C. Wang, F. Bi, W. Zhang, and L. Chen, "An intensity-space domain CFAR method for ship detection in HR SAR images," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 4, pp. 529–533, Apr. 2017.

[6] T. Li, Z. Liu, R. Xie, and L. Ran, "An improved superpixel-level CFAR detection method for ship targets in high-resolution SAR images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 1, pp. 184–194, Jan. 2018.

[7] H. Greidanus, M. Alvarez, C. Santamaria, F.-X. Thoorens, N. Kourti, and P. Argentieri, "The SUMO ship detector algorithm for satellite radar images," *Remote Sens.*, vol. 9, no. 3, p. 246, 2017.

[8] C. Wang, S. Jiang, H. Zhang, F. Wu, and B. Zhang, "Ship detection for high-resolution SAR images based on feature analysis," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 1, pp. 119–123, Jan. 2014.

[9] C. P. Schwegmann, W. Kleynhans, and B. P. Salmon, "Synthetic aperture radar ship detection using Haar-like features," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 2, pp. 154–158, Feb. 2017.

[10] W. Ao, F. Xu, Y. Li, and H. Wang, "Detection and discrimination of ship targets in complex background from spaceborne ALOS-2 SAR images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 2, pp. 536–550, Feb. 2018.

[11] M. Tello, C. López-Martínez, and J. J. Mallorquí, "A novel algorithm for ship detection in SAR imagery based on the wavelet transform," *IEEE Geosci. Remote Sens. Lett.*, vol. 2, no. 2, pp. 201–205, Apr. 2005.

[12] A. Marino, M. J. Sanjuan-Ferrer, I. Hajnsek, and K. Ouchi, "Ship detection with spectral analysis of synthetic aperture radar: A comparison of new and well-known algorithms," *Remote Sens.*, vol. 7, no. 5, pp. 5416–5439, 2015.

[13] R. Pelich, N. Longépé, G. Mercier, G. Hajduch, and R. Garello, "AIS-based evaluation of target detectors and SAR sensors characteristics for maritime surveillance," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 8, pp. 3892–3901, Aug. 2015.

[14] P. Salembier, A. Oliveras, and L. Garrido, "Motion connected operators for image sequences," in *Proc. 8th Eur. Signal Process. Conf. (EUSIPCO)*, Trieste, Italy, Sep. 1996, pp. 1083–1086.

[15] P. Salembier, A. Oliveras, and L. Garrido, "Antiextensive connected operators for image and sequence processing," *IEEE Trans. Image Process.*, vol. 7, no. 4, pp. 555–570, Apr. 1998.

[16] R. Jones, "Component trees for image filtering and segmentation," in *Proc. IEEE Workshop Nonlinear Signal Image Process.*, Mackinac Island, MI, USA, 1997, pp. 1–5.

[17] P. Monasse and F. Guichard, "Fast computation of a contrast-invariant image representation," *IEEE Trans. Image Process.*, vol. 9, no. 5, pp. 860–872, May 2000.

[18] N. Ray and S. T. Acton, "Inclusion filters: A class of self-dual connected operators," *IEEE Trans. Image Process.*, vol. 14, no. 11, pp. 1736–1746, Nov. 2005.

[19] P. Salembier and L. Garrido, "Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval," *IEEE Trans. Image Process.*, vol. 9, no. 4, pp. 561–576, Apr. 2000.

[20] P. Soille, "Constrained connectivity for hierarchical image partitioning and simplification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 7, pp. 1132–1145, Jul. 2008.

[21] C. Berger, T. Geraud, R. Levillain, N. Widynski, A. Baillard, and E. Bertin, "Effective component tree computation with application to pattern recognition in astronomical imaging," in *Proc. Int. Conf. Image Process.*, San Antonio, TX, USA, Sep./Oct. 2007, pp. IV-41–IV-44.

[22] V. Vilaplana, F. Marques, and P. Salembier, "Binary partition trees for object detection," *IEEE Trans. Image Process.*, vol. 17, no. 11, pp. 2201–2216, Nov. 2008.

[23] G. K. Ouzounis and M. H. F. Wilkinson, "Hyperconnected attribute filters based on k-flat zones," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 2, pp. 224–239, Feb. 2011.

[24] E. J. Breen and R. Jones, "Attribute openings, thinnings, and granulometries," *Comput. Vis. Image Understand.*, vol. 64, no. 3, pp. 377–389, Nov. 1996.

[25] P. Salembier and M. H. F. Wilkinson, "Connected operators: A review of region-based morphological image processing techniques," *IEEE Signal Process. Mag.*, vol. 26, no. 6, pp. 136–157, Nov. 2009.

[26] H. Lu, J. C. Woods, and M. Ghanbari, "Binary partition tree analysis based on region evolution and its application to tree simplification," *IEEE Trans. Image Process.*, vol. 16, no. 4, pp. 1131–1138, Apr. 2007.

[27] Y. Xu, E. Carlinet, T. Géraud, and L. Najman, "Hierarchical segmentation using tree-based shape spaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 3, pp. 457–469, Mar. 2017.

[28] G.-S. Xia, J. Delon, and Y. Gousseau, "Shape-based invariant texture indexing," *Int. J. Comput. Vis.*, vol. 88, no. 3, pp. 382–403, Jul. 2010.

[29] C. Kurtz, N. Passat, P. Gançarski, and A. Puissant, "Extraction of complex patterns from multiresolution remote sensing images: A hierarchical top-down methodology," *Pattern Recognit.*, vol. 45, no. 2, pp. 685–706, 2012.

[30] A. Alonso-Gonzalez, S. Valero, J. Chanussot, C. Lopez-Martinez, and P. Salembier, "Processing multidimensional SAR and hyperspectral images with binary partition tree," *Proc. IEEE*, vol. 101, no. 3, pp. 723–747, Mar. 2013.

[31] M. A. Veganzones, G. Tochon, M. Dalla-Mura, A. J. Plaza, and J. Chanussot, "Hyperspectral image segmentation using a new spectral unmixing-based binary partition tree representation," *IEEE Trans. Image Process.*, vol. 23, no. 8, pp. 3574–3589, Aug. 2014.

[32] M. A. Westenberg, J. B. T. M. Roerdink, and M. H. F. Wilkinson, "Volumetric attribute filtering and interactive visualization using the max-tree representation," *IEEE Trans. Image Process.*, vol. 16, no. 12, pp. 2943–2952, Dec. 2007.

[33] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Process. Mag.*, vol. 30, no. 3, pp. 83–98, May 2013.

[34] A. Sandryhaila and J. M. F. Moura, "Big data analysis with signal processing on graphs: Representation and processing of massive data sets with irregular structure," *IEEE Signal Process. Mag.*, vol. 31, no. 5, pp. 80–90, Sep. 2014.
[35] S. Chen, R. Varma, A. Sandryhaila, and J. Kovačević, "Discrete signal processing on graphs: Sampling theory," *IEEE Trans. Signal Process.*, vol. 63, no. 24, pp. 6510–6523, Dec. 2015.
[36] S. K. Narang and A. Ortega, "Perfect reconstruction two-channel wavelet filter banks for graph structured data," *IEEE Trans. Signal Process.*, vol. 60, no. 6, pp. 2786–2799, Jun. 2012.
[37] A. Sakiyama and Y. Tanaka, "Oversampled graph Laplacian matrix for graph filter banks," *IEEE Trans. Signal Process.*, vol. 62, no. 24, pp. 6425–6437, Dec. 2014.
[38] D. Shuman *et al.*, "A multiscale pyramid transform for graph signals," *IEEE Trans. Signal Process.*, vol. 64, no. 8, pp. 2119–2134, Apr. 2013.
[39] J. Cousty, L. Najman, F. Dias, and J. Serra, "Morphological filtering on graphs," *Comput. Vis. Image Understand.*, vol. 117, no. 4, pp. 370–385, 2013.
[40] Y. Xu, T. Géraud, and L. Najman, "Connected filtering on tree-based shape-spaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 6, pp. 1126–1140, Jun. 2016.
[41] S. Valero, P. Salembier, and J. Chanussot, "Object recognition in hyperspectral images using Binary Partition Tree representation," *Pattern Recognit. Lett.*, vol. 56, pp. 45–51, Apr. 2015.
[42] E. Carlinet and T. Géraud, "A comparative review of component tree computation algorithms," *IEEE Trans. Image Process.*, vol. 23, no. 9, pp. 3885–3895, Sep. 2014.
[43] Y. Hu, "Efficient, high-quality force-directed graph drawing," *Math. J.*, vol. 10, no. 1, pp. 37–71, 2005.
[44] *Graphviz—Graph Visualization Software*. Accessed: 2018. [Online]. Available: https://www.graphviz.org
[45] A. Sandryhaila and J. M. F. Moura, "Discrete signal processing on graphs," *IEEE Trans. Signal Process.*, vol. 61, no. 7, pp. 1644–1656, Apr. 2013.
[46] S. Chen, A. Sandryhaila, J. M. F. Moura, and J. Kovacevic, "Signal denoising on graphs via graph filtering," in *Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP)*, Dec. 2014, pp. 872–876.
[47] S. Segarra, A. G. Marques, G. R. Arce, and A. Ribeiro, "Center-weighted median graph filters," in *Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP)*, Dec. 2016, pp. 336–340.
[48] *Maxtree Processing Toolbox*. Accessed: 2018. [Online]. Available: https://github.com/imatge-upc/Maxtree-Processing-Toolbox
[49] C. Vachier and F. Meyer, "Extinction value: A new measurement of persistence," in *Proc. IEEE Workshop Nonlinear Signal Image Process. (NSIP)*, Halkidiki, Greece, vol. 1, Jun. 1995, pp. 254–257.
[50] A. G. Silva and R. D. A. Lotufo, "New extinction values from efficient construction and analysis of extended attribute component tree," in *Proc. 21st Brazilian Symp. Comput. Graph. Image Process.*, Oct. 2008, pp. 204–211.
[51] X. Wang and C. Chen, "Ship detection for complex background SAR images based on a multiscale variance weighted image entropy method," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 2, pp. 184–187, Feb. 2017.
[52] M. J. Sanjuan-Ferrer, "Detection of coherent scatterers in SAR data: Algorithms and applications," Ph.D. dissertation, ETH Zurich, Zürich, Switzerland, 2013.
[53] M. T. Alonso, C. López-Martínez, J. J. Mallorquí, and P. Salembier, "Edge enhancement algorithm based on the wavelet transform for automatic edge detection in SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 1, pp. 222–235, Jan. 2011.
[54] F. Baselice and G. Ferraioli, "Unsupervised coastal line extraction from SAR images," *IEEE Geosci. Remote Sens. Lett.*, vol. 10, no. 6, pp. 1350–1354, Nov. 2013.
[55] M. Modava and G. Akbarizadeh, "Coastline extraction from SAR images using spatial fuzzy clustering and the active contour method," *Int. J. Remote Sens.*, vol. 38, no. 2, pp. 355–370, 2017.
[56] G. Akbarizadeh, "A new statistical-based kurtosis wavelet energy feature for texture recognition of SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 11, pp. 4358–4368, Nov. 2012.
[57] K. Kayabol and J. Zerubia, "Unsupervised amplitude and texture classification of SAR images with multinomial latent model," *IEEE Trans. Image Process.*, vol. 22, no. 2, pp. 561–572, Feb. 2013.
[58] Q. Oliveau and H. Sahbi, "Learning attribute representations for remote sensing ship category classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 6, pp. 2830–2840, Jun. 2017.

Philippe Salembier (M'96–SM'09–F'11) received the Electrical Engineering degrees from Ecole Polytechnique, Palaiseau, France, and Ecole Nationale Supérieure des Télécommunications, Paris, France, in 1983 and 1985, respectively, and the Ph.D. from the EPFL, Lausanne, Switzerland, in 1991.

From 1985 to 1989, he was with the Laboratoires d'Electronique Philips, Limeil-Brévannes, France. In 1989, he joined the Swiss Federal Institute of Technology in Lausanne, Lausanne. He was a Post-Doctoral Fellow with the Harvard Robotics Laboratory, Cambridge, MA, USA. In 1992, he joined the Technical University of Catalonia–BarcelonaTech, Barcelona, Spain, where he is currently a Professor of digital signal and image processing. His research interests include remote sensing image processing, image and video sequence processing, and mathematical morphology.

Dr. Salembier was a member of the Image and Multidimensional Signal Processing Technical Committee of the IEEE Signal Processing Society between 2000 and 2006 and an AdCom Officer of the European Association for Signal Processing (EURASIP) between 1994 and 1999. He was the Chair of the Multimedia Description Scheme Group from 1999 to 2001, when he was involved in the definition of the MPEG-7 standard (Multimedia Content Description Interface). He served as an Associate Editor of various journals including the *Journal of Visual Communication and Image Representation* (Academic Press), *Signal Processing* (Elsevier), *Signal Processing: Image Communication* (Elsevier), the *EURASIP Journal on Image and Video Processing*, the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, and the IEEE SIGNAL PROCESSING LETTERS.

Sergi Liesegang received the bachelor's degree in telecommunication engineering from ETSETB, Technical University of Catalonia–BarcelonaTech, Barcelona, Spain, in 2015, and the master's degree in telecommunication engineering from the Technical University of Catalonia–BarcelonaTech, in collaboration with the Technical University of Munich, Munich, Germany, in 2017. He is currently pursuing the Ph.D. degree with the Signal Theory and Communications Department, Technical University of Catalonia–BarcelonaTech.

From 2015 to 2017, he was a Research Assistant with Signal Theory and Communications Department. His research interests include signal processing, multimedia technologies, and information and communication theory.

Carlos López-Martínez (S'97–M'04–SM'11) received the MSc. degree in electrical engineering and the Ph.D. degree from the Universitat Politècnica de Catalunya, Barcelona, Spain, in 1999 and 2003, respectively.

From 2000 to 2002, he was with the Frequency and Radar Systems Department, German Aerospace Center, Oberpfaffenhofen, Germany. From 2003 to 2005, he has been with the Image and Remote Sensing Group—SAPHIR Team, Institute of Electronics and Telecommunications of Rennes, Rennes, France. In 2006, he joined the Technical University of Catalonia–BarcelonaTech, Barcelona, Spain, as a Ramón-y-Cajal Researcher, where he is currently an Associate Professor of remote sensing and microwave technology. He has authored or co-authored more than 100 articles in journals, books, and conference proceedings in the radar remote sensing and image analysis literature. His research interests include synthetic-aperture radar (SAR) and multidimensional SAR, radar polarimetry, physical parameter inversion, digital signal processing, estimation theory, and harmonic analysis.

Dr. López-Martínez has organized different invited sessions in international conferences on radar and SAR polarimetry. He has presented advanced courses and seminars on radar polarimetry to a wide range of organizations and events. He was a recipient of the Student Prize Paper Award at the EUSAR 2002 Conference, the First Place Student Paper Award at the EUSAR 2012 Conference, and the IEEE-GRSS 2013 GOLD Early Career Award. He served as a Guest Editor for the *European Association for Signal Processing Journal on Advances in Signal Processing*. He is an Associate Editor of the IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING.